WHAT IS CLAIMED IS:

1. An adder based circuit embodied in an integrated circuit comprising

an input module responsive to inputs A[i] and B[i] to generate a first input function U[i] = A[i]&B[i] and a second input function V[i] = A[i]∨B[i] or V[i] = A[i]⊕B[i];

a carry module responsive to the first and second input functions to generate carry functions; and

an output module responsive to the first and second input functions and the carry function to provide an output function $S=\sum_{i=0}^{n-1}2^{i}S[i]$ = $\sum_{i=0}^{n-1}2^{i}A[i]$ + $\sum_{i-0}^{n-1}2^{i}B[i]$,

wherein the carry module has a minimal depth defined by a recursive expansion of at least one function associated with the carry module based at least in part on a variable k, where k = F_l and n-k = F_{l-1} and where l satisfies F_l < n = F_{l+1}, {F_l} is a Fibonacci series and n is the number of bits of at least one of the inputs.

2. The adder based circuit of claim 1, including one or more additional elements coupled to at least one of the inputs of the input module and outputs of the output module to perform the function selected from the group comprising subtractors,

adder-subtractors, incrementors, decrementors, increment-decrementors and absolute value calculators.

3. The adder based circuit of claim 2, wherein the additional elements are selected from the group comprising inverter elements, exclusive or elements, exclusive nor elements and multiplexers.

4. A process of designing an adder based circuit for an integrated circuit comprising steps of:

a) defining at least one carry function of a carry module of the adder based circuit in terms of a Fibonacci series; and

b) recursively expanding the carry function to find a minimum parameter of the Fibonacci series.

5. The process of claim 4, wherein step (b) comprises steps of:

b1) defining recursive functions

h'_l = h_1(U[k+1],U[k+2],V[k+2],...,
U[k+1],V[k+1]) and

v'_l = V[k+1]&,...,&V[k+1],

where l = 1,...,n-k and U[i] and V[i] are inputs to the carry module, k=F_l and n-k=F_{l-1}, l satisfies

F_l<n=F_{l+1}, {F_l} is the Fibonacci series defined recursively from the equality F_{l+1}=F{l}+F_{l+1} and n is the number of bits of an input to the carry module, and

b2) recursively expanding the recursive functions to minimize l.

6. The process of claim 5, wherein the recursive functions are selected from the group consisting of HV_{n-k} = {h'_1,v'_1,...,h'_{n-k},v'_{n-k,}} and either (i) H_k = {h_1,..., h_k} or (ii) HV_k = h_1,v_1,...,h_k,v_k and HV_{n-k} = h'_1,v'_1,...,h'_{n-k},v'_{n-k}, where v_k = V_n]&...&V[n-k+1].

7. The process of claim 4, further including steps of:

c) optimizing a distribution of negations in the carry module.

8. The process of claim 7, wherein step (c) comprises, for each portion of the carry module, steps of:

c1) identifying a set of delay vectors,

c2) recursively comparing the delay vectors to derive a set of minimum vectors, and

c3) selecting a vector with a minimum norm from the set of minimum vectors.

9. The process of claim 4, further including a step of:

c) minimizing a depth of the adder based circuit.

10. The process of claim 9, wherein step (c) comprises:

c1) recursively expanding expressions

DH_n = min{max(DH_{n-k}+1,Dh_k+2,DH_k}} and

Dh_n = min{max(Dh_{n-k}+1,Dh_k+2}}

where DH_k and Dh_k are based on vectors of depths U[i] for $0 \leq i \leq k-1$ and DH_{n-k} and Dh_{n-k} are based on vectors for $k \leq i \leq n-1$, and

c2) selecting a value of k based on a minimum of at least one of DH_n and Dh_n.

11. The process of claim 10, wherein step (c2) comprises steps of

calculating vectors DHh_n based on maximum values of DH_n and Dh_n,

sorting values of k to derive a set of minimum vectors DHh_n, and

searching the set of minimum vectors to select a value of k.

12. The process of claim 4, further including a step of:

c) minimizing fanout depth of the adder based circuit.

13. The process of claim 12, wherein step (c) comprises steps of:

c1) defining recursive functions

$$H^i_k = h^{i+1}_{k-1} \vee v^{i+1}_{k-1} \& h^i_1$$

and

$$v^i_1 = v^{i+1}_{k^1} \& v^i_1$$

based on a function of the adder based circuit, where $k=F_1$ and $n-k=F_{1-1}$, 1 satisfies $F_1<n=F_{1+1}$, $\{F_1\}$ is the Fibonacci series defined from the equality $F_{1+1}=F\{1\}+F_{1+1}$ and n is the number of bits of an input to the adder based circuit, and

c2) recursively expanding the recursive functions to minimize 1.

14. A storage medium having stored therein

first processor executable instructions that enable a processor to define an adder based circuit structure for an integrated circuit comprising elements that define at least one output function in terms of a Fibonacci series; and

second processor executable instructions that enable the processor to recursively expand the output function to find a minimum parameter of the Fibonacci series.

15. The storage medium of claim 14, wherein the second processor executable instructions comprises:

processor executable instructions that enable the processor to define recursive functions h'_l = h_l(U[k+1],U[k+2],V[k+2],..., U[k+1],V[k+1]) and v'_l = V[k+1]&,...,&V[k+1], based on the carry function, where k=F_l and n-k=F_{l-1}, l satisfies F_l<n=F_{l+1}, {F_l} is the Fibonacci series defined recursively from the equality F_{l+1}=F{l}+F_{l+1} and n is the number of bits of an input to the carry module, and

processor executable instructions that enable the processor to recursively expand the recursive functions to minimize l.

16. The storage medium of claim 15, wherein the recursive functions are selected from the group consisting of HV_{n-k} = {h'_1,v'_1,...,h'_{n-k},v'_{n-k,}} and either (i) H_k = {h_1,..., h_k} or (ii) HV_k = h_1,v_1,...,h_k,v_k and HV_{n-k} = h'_1,v'_1,...,h'_{n-k},v'_{n-k}, where v_k = V_n]&...&V[n-k+1].

17. The storage medium of claim 14, further including third processor executable instructions that enable the processor to optimize a distribution of negations in the carry module, the third processor executable instructions comprising:

processor executable instructions that enable the processor to identify a set of delay vectors in each portion of the carry module,

processor executable instructions that enable the processor to recursively compare the delay vectors in each portion to derive a set of minimum vectors, and

processor executable instructions that enable the processor to select a vector for each portion with a minimum norm from the set of minimum vectors.

18. The storage medium of claim 14, further including third processor executable instructions that enable the processor to minimize depth of the adder base circuit, the third processor executable instructions comprising:

processor executable instructions that enable the processor to recursively expand expressions

DH_n = min{max(DH_{n-k}+1,Dh_k+2,DH_k}} and

Dh_n = min{max(Dh_{n-k}+1,Dh_k+2}}

where DH_k and Dh_k are based on vectors of depths U[i] for $0 \leq i \leq k-1$ and DH_{n-k} and Dh_{n-k} are based on vectors for $k \leq i \leq n-1$, and

processor executable instructions that enable the processor to selecting a value of k based on a minimum of at least one of DH_n and Dh_n.

19. The storage medium of claim 18, wherein the computer instructions that enable the processor to select a value of k comprises:

processor executable instructions that enable the processor to calculate vectors DHh_n based on maximum values of DH_n and Dh_n,

processor executable instructions that enable the processor to sort values of k to derive a set of minimum vectors DHh_n, and

processor executable instructions that enable the processor to search the set of minimum vectors to select a value of k.

20. The storage medium of claim 14, further including third processor executable instructions that enable the processor to minimize fanout depth of the adder base circuit, the third processor executable instructions comprising:

processor executable instructions that enable the processor to define recursive functions

$H^i_k = h^{i+1}_{k-1}vv^{i+1}_{k-1}\&h^i_l$

and

$v^i_l = v^{i+1}_{k^l}\&v^i_l$

based on a function of the adder based circuit, where $k=F_l$ and $n-k=F_{l-1}$, l satisfies $F_l<n=F_{l+1}$, $\{F_l\}$ is the Fibonacci series defined from the equality $F_{l+1}=F{l}+F_{l+1}$ and n is the number of bits of an input to the adder based circuit, and

processor executable instructions that enable the processor to recursively expand the recursive functions to minimize l.